

CLAIMS

1. A method for testing visibility of graphics primitives, which method comprises the steps of:
 - computing the geometry of graphics primitives;
 - 5 testing the visibility of the computed primitives in the first visibility test;
 - based on said first test storing the occlusion data of the visible primitives for next comparison;
 - and
 - 10 computing the occlusion culling data for each visible primitive;
 - characterized in that the method further comprises steps:
 - collecting stored primitives to an occlusion culling data buffer;
 - 15 testing the visibility of the collected primitives in the second visibility test;
 - rasterizing visible primitives of the second visibility test.
- 20 2. The method according to claim 1, characterized in that discarding the hidden primitives of the first visibility test.
3. The method according to claim 1 or 2, characterized in that storing Z values to occlusion fusion cache while computing occlusion.
- 25 4. The method according to claim 1, 2 or 3, characterized in that after said first test collecting occlusion data of the visible primitives belonging to the frame to be rendered to the occlusion culling data buffer.
- 30 5. The method according to claim 1- 2 or 3, characterized in that after said first test collecting a predefined amount of occlusion data of the primitives to the occlusion culling data buffer.
- 35 6. The method according any of preceding claims 1 - 5, characterized in that compressing the occlusion buffer.

7. The method according any of preceding claims 1 - 6, characterized in that the method further comprises testing visibility of the object before the geometry processor by bounding volume method.

8. The method according to any of preceding claims 1 - 7, characterized in that testing the visibility of the primitive in the first and the second visibility test with low resolution Z-buffer.

9. A system for testing visibility of graphics primitives, which system further comprises:

a Geometry processor (20);
a Z-buffer component (21);
first visibility test module (22);
occlusion fusion unit (23); and
pixel processing means (26)
characterized in that the system further comprises:

an occlusion data buffer (24); and
a second visibility test module (25);

10. The system according to claim 9, characterized in that the first visibility test is arranged (22) to discard hidden primitives.

11. The system according to claims 9 or 10, characterized in that the occlusion data buffer (24) is arranged to collect occlusion data of the primitives belonging to the frame to be rendered.

12. The system according to claims 9 or 10, characterized in that occlusion data buffer (24) is arranged to collect a predefined amount of occlusion data of the primitives.

13. The system according to any of preceding claims 9 - 12, characterized in that the system further comprises means for compressing (29) and decompressing (212) the occlusion data buffer (24).

14. The system according to any preceding claims 9 - 13, characterized in that the sys-

tem further comprises means for bounding volume testing.

15. The system according to any preceding claim 9 - 14, characterized in that the system further comprises an occlusion fusion cache.

16. The system according to any preceding claim 9 - 15, characterized in that the Z-buffer connected to first visibility test module is a low resolution Z-buffer.

17. The system according to claim 17, characterized in that the system further comprises a high resolution Z-buffer connected to said second visibility test.

18. The system according to claim 16, characterized in that the values stored to the low resolution Z-buffer are calculated in occlusion fusion cache.